

# The fascinating story of core-js | Frontend Weekly

## vol. 125

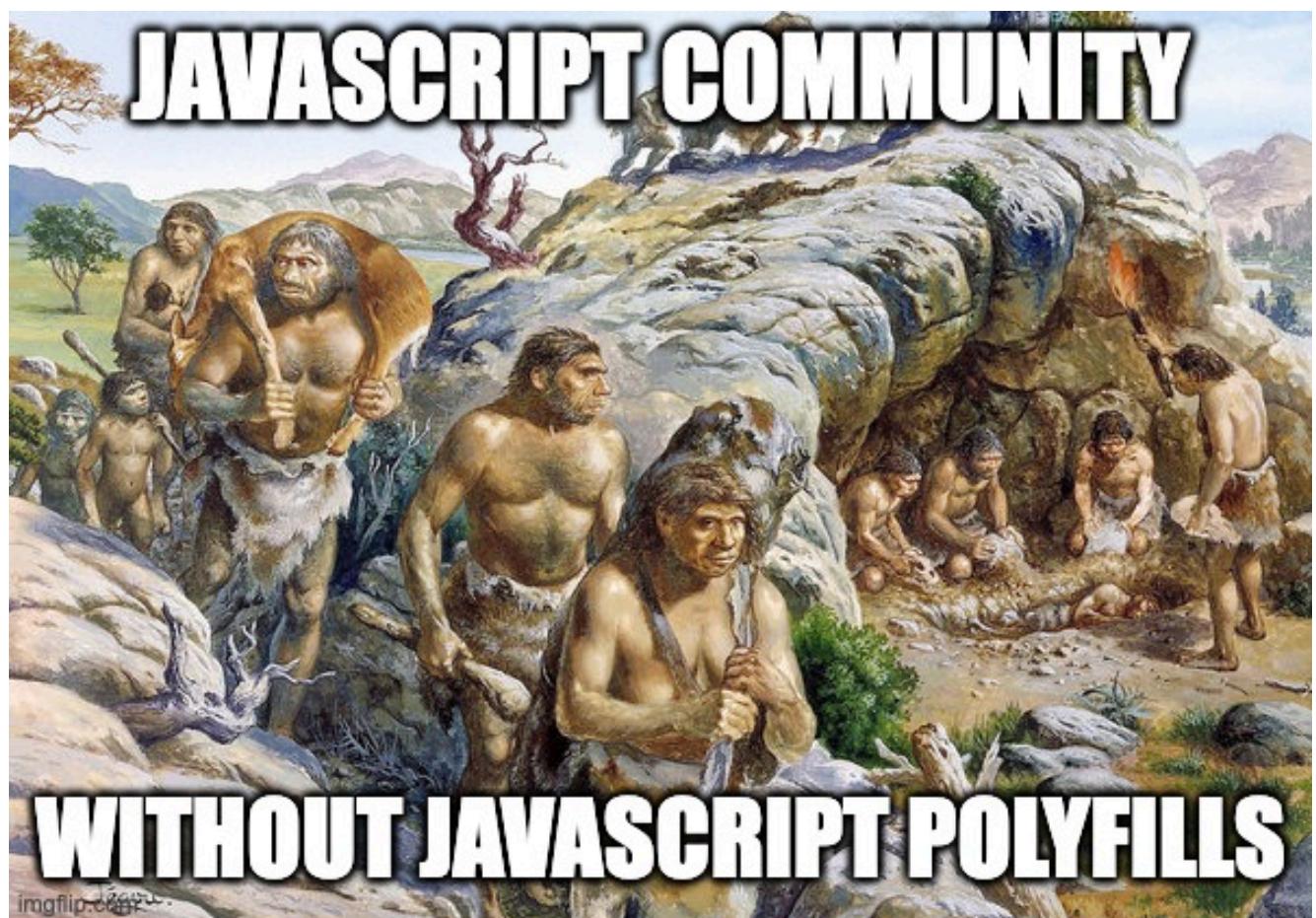


It's a well-known fact that open source is hard work. For a typical library maintainer, every day there are a lot of requests for new functionalities and even more bug reports. Not to mention that there are more and more people dissatisfied with the pace of development of the project every day. Interest does not translate into contributions or financial support. Every few months the tide of bitterness overflow for another developer. This week it overflowed for the core-js maintainer.

## The intriguing story of core-js

`core-js` is the most popular library that provides polyfills for JavaScript. A polyfill is an adapter that allows modern code to run in old browsers. If it weren't for polyfills, we'd still be doomed to write in archaic ES5.

How popular is `core-js`? From tests conducted by the library maintainer, out of the 1000 most popular websites, more than 500 use `core-js`. The number of companies dependent on `core-js` certainly exceeds 50%. Even if these largest companies' applications are not dependent on the library, there are at least a few smaller tools in the company's portfolio that have such a dependency.



As it usually happens, life writes the best scenarios. In 2012, Denis Pushkarev became a full-stack developer and began to deal with JavaScript on a daily basis. Frustrated

by the slow adoption of the ES6 standard, he began to implement the necessary polyfills on his own. In late 2014, he decided to publish his package under the name `core-js`. Soon after, this package became one of the most important dependencies of Babel and several other popular frameworks.

On the wave of success of his library, Denis Pushkarev decided to devote himself fully to open source. For the time being, the project did not have adequate funding, but the author hoped that if he did a good job, the money would come sooner or later. Either in the form of donations or the form of some big company hiring him so he could devote himself to the project. Weeks passed and the project budget did not grow. That's when Denis Pushkarev made a decision that would determine his future. In order to cut living costs and continue developing `core-js`, he decided to return to his motherland Russia.

Shortly after returning to Russia, a real tragedy happened. Late in the night, the author `core-js` was returning home by car. Two drunken women wearing dark clothes appeared in his way. According to witnesses, the girls had been fooling around and fighting. Denis Pushkarev failed to react in time – one of the girls died, while the other was taken to the hospital.

Unfortunately, Russian law did not work in favor of the driver in this situation. Despite the fact that no pedestrians should be on the road at the place of the

accident, the fault was clearly placed on the side of the driver. He faced up to several years in prison. The only reasonable way out of the situation was a settlement with the families of the accident victims. Negotiations resulted in a sum of \$80k plus legal costs. For Denis Pushkarev, this was a big amount of money. There was nothing left for him to do but to try to finally monetize `core-js`.

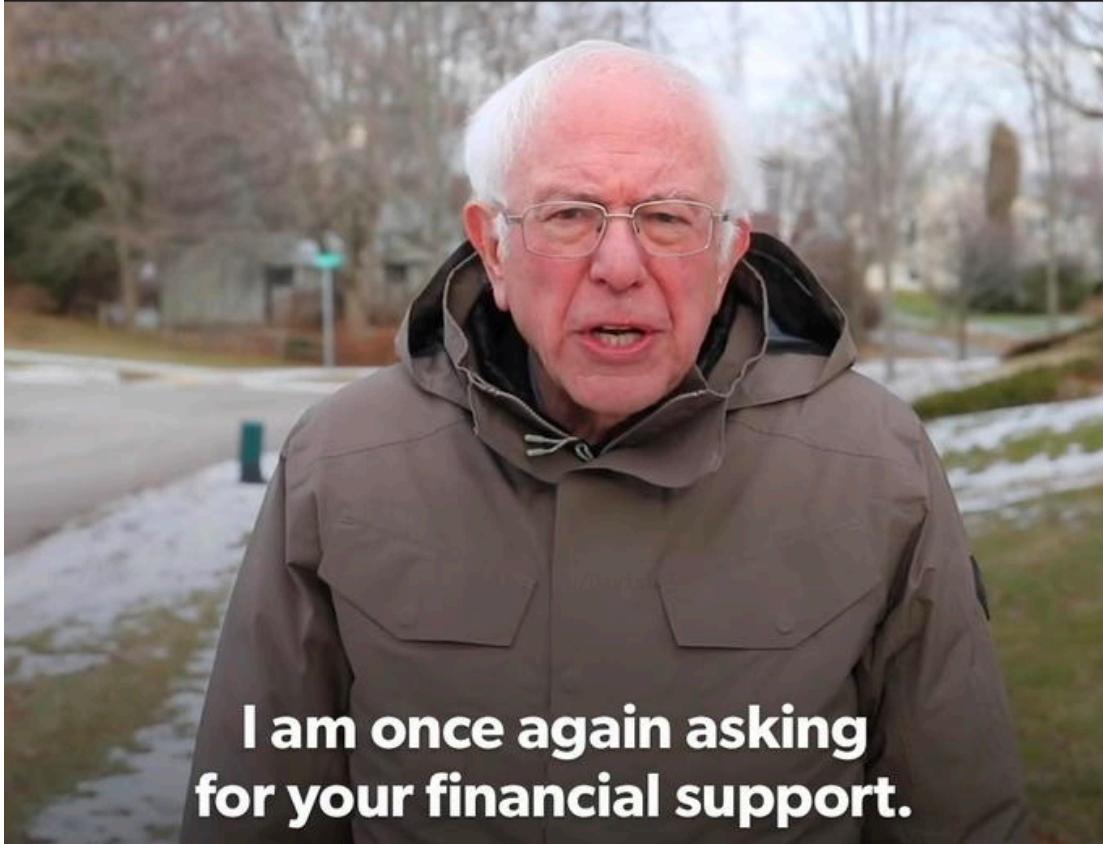
The first idea was to display a prompt for support when installing `core-js`. The community's response fell short of the author's expectations. Instead of financial support, he received a rash of hateful comments. The next attempt was to properly configure `npm` `found`. As you may have guessed, this also did not have the intended effect. After all – have you ever used or even heard of this functionality? As a final attempt, the `core-js` author included a job announcement in the installation prompt. This did not have the intended effect too.

```
Thank you for using core-js ( https://github.com/zloirock/core-js ) for polyfilling JavaScript standard library!
The project needs your help! Please consider supporting of core-js on Open Collective or Patreon:
> https://opencollective.com/core-js
> https://www.patreon.com/zloirock

Also, the author of core-js ( https://github.com/zloirock ) is looking for a good job -)
```

```
$ npm install
```

```
core-js:
```



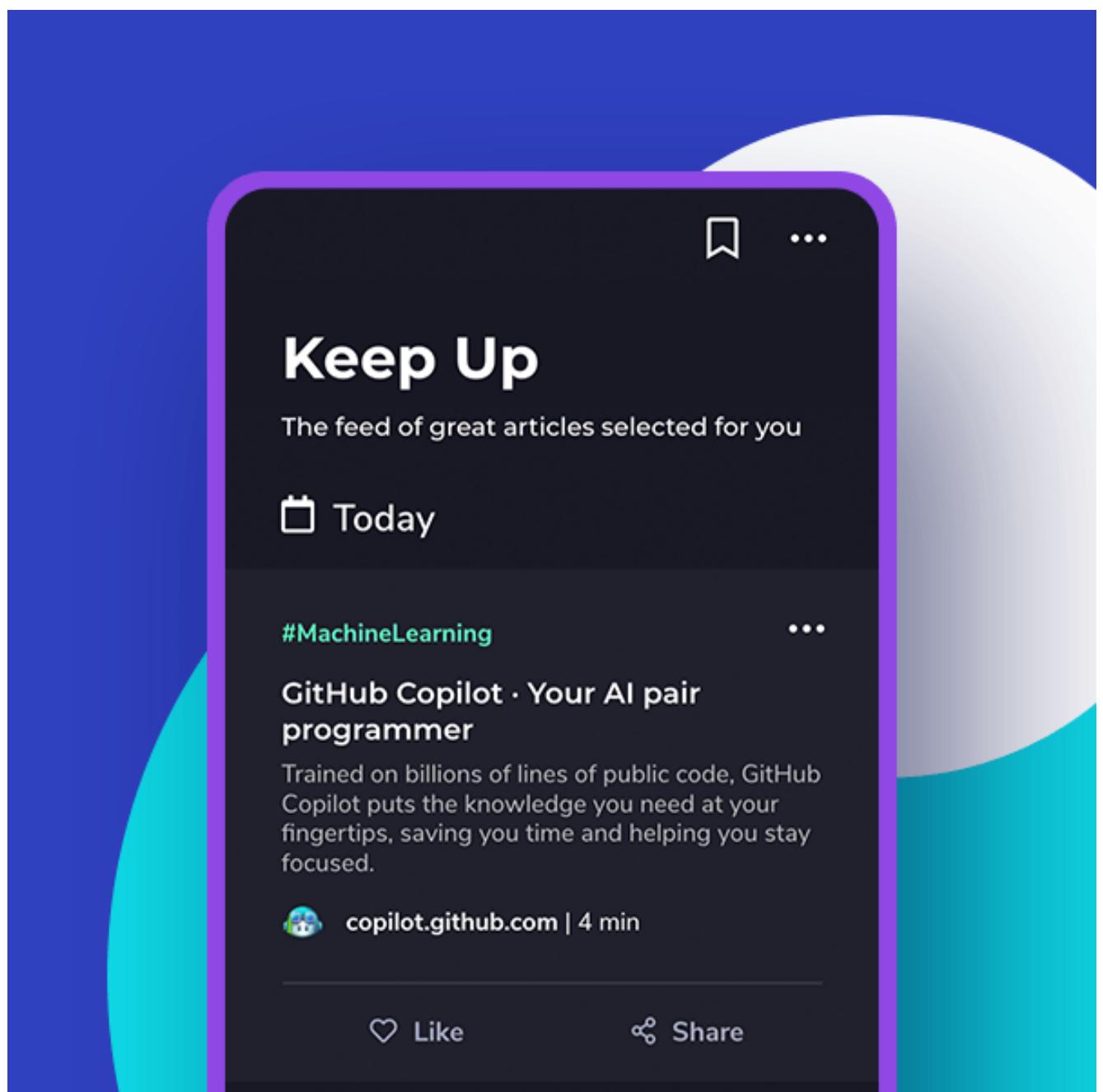
The necessary amount was not raised in time, and in January 2020 Denis Pushkarev was sent to prison. For 10 months he was forced to work in a chemical factory among real criminals. After this period he was conditionally released and returned to work on core-js.

It has recently been two years since Denis Pushkarev left prison, and the core-js situation has still not improved. To work on the library, the author collects about \$2k per month. Enough to make a living in Russia but compared to other developers employed by large companies – still far too little.

This week something snapped in Denis Pushkarev, and he made the following statement: either there will be money for core-js development or he will abandon the project altogether.

Discover more IT content selected for you

In Vived, you will find articles handpicked by devs. Download the app and read the good stuff!



**And what does it look like with other**

# open-source libraries?

At the beginning of 2021, Babel maintainers announced that they are running out of money. Despite the initial success of the Open Collective fundraiser, less and less money was flowing into the developers' accounts. The initial goal of creating a 4-person team was drifting further and further away, and maintaining the current 3-person team was becoming increasingly difficult. The fuss surrounding the project caused a temporary improvement, but after two years the project's monthly budget returned to its baseline. True, Babel is supported by several large companies (such as AirBnB, GitHub, or Salesforce), but the amount of \$16k per month is not impressive for a project that is the foundation of the modern Internet.

In mid-2021, Marak Squires, an author of many popular npm libraries, published a post titled "Monetizing Open Source is Problematic". Around the same time, Marak added a "No more free work from Marak – Pay Me or Fork This" Issue to his most popular `faker.js` library. Six months later, Marak, frustrated with the situation, decided to remove the `faker.js` from the internet. A few weeks later, he intentionally introduced a critical bug to the `color.js` library and then submitted an ironic Issue describing how he was hard working to fix it. As you might expect, the community's reaction to such radical steps was.... mixed. Some supported the developer's extremist

moves and granted him full rights to his code. Others argued that such actions are toxic and do not bring us any closer to solving the problems.

The problems with open source are not just in our JavaScript community. In 2015, Steve Marquess resigned from further development of OpenSSL – one of the critical parts of the current web infrastructure. In this case, it was not about financial issues, but about unrealistic expectations for a project maintained by one person. Another high-profile story from the open-source community is the departure of Sarah Sharp from the Linux kernel project. In this case, it was about the toxic community created around the project and resistance to the introduction a proper Code of Conduct.

There is also the bright side of open-source development. Some developers are trying to build start-ups around their projects. The list is really long, so I will cite just a few examples to support my thesis: Next.js (4 funding rounds of over \$300M), Deno (2 funding rounds of over \$20M), Bun (1 funding round of over \$7M) or Prisma (2 funding rounds of over \$50M). In the past year, quite a few such start-ups have found shelter under the wings of larger companies. Just to mention the recent acquisition of Gatsby by Netlify, Remix by Shopify, and Ionic by Outsystems.

Another trend we've seen in recent years is that large companies are hiring authors of promising libraries to

work full-time on them full. This is what Vercel did with the developer of Svelte and Netlify with the maintainers of Solid.js and Eleventy. Such moves usually have a solid business background. Vercel provides a network infrastructure ideal for hosting SvelteKit. Both Solid.js and Eleventy fit perfectly into JAMStack, Netlify's core business area.

**You can read more about the aforementioned acquisitions in:**

[Bun is baked very quickly – Frontend Weekly vol. 102](#)

[Shopify went shopping. What's next for Remix and Hydrogen? | Frontend Weekly vol. 111](#)

[Gatsby counterattacks | Frontend Weekly vol. 112](#)

[Netlify acquires Gatsby | Frontend Weekly vol. 123](#)

It is worth remembering that not every worthwhile project can be turned into a start-up, and not every project will find a company where it will fit well. History shows that financial support from library clients is usually not something we can count on.

**What will happen if core-js disappear?**

Does the above meme by “XKCD” have anything to do with reality? A great example that comes to mind is the story of the 11-line `left-pad` library. 28-year-old developer Azer Koçulu was one of the many Open Source developers firmly committed to the traditional values of the community. One of the libraries he maintained for

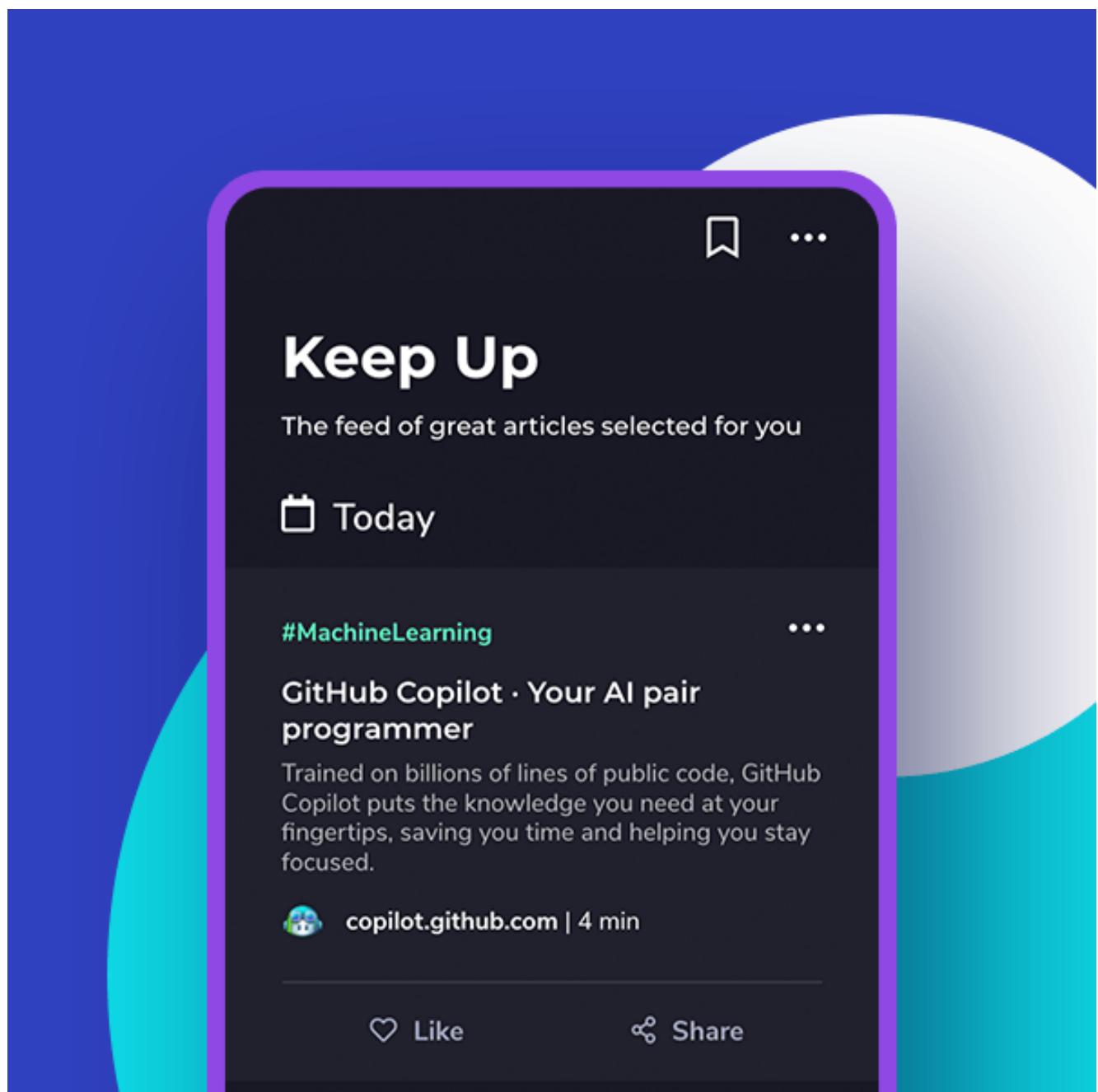
years was called "kik". In March 2016, he was approached by the newly formed company "kik" asking him to rename his package in npm. The conversation went from requests to financial negotiations and then to legal threats. The frustrated developer decided to remove his libraries from the Internet altogether. That's how, in late March, developers around the world began to face the mysterious 'left-pad' is not in the npm registry error. Most of the developers had never heard of `left-pad`, and yet it somehow messed up millions of applications. As you may have guessed – `left-pad` was a dependency of our dependencies. Such a popular dependency that its removal crashed builds of most applications on the Internet. It was also the first time when npm restored a deleted library.

In `core-js` case, the library could be suspended but not completely removed from the internet. The ECMAScript standard will continue to develop rapidly. Without `core-js` we will not be able to adapt new functionalities so quickly. Maybe someone will take over the development of the library? That's not likely to happen either. In 2020, Babel's developers communicated that they did not have the resources to take over the project. If not them, it's hard to find another team with the right skills. Maybe we could just switch to some reasonable alternatives? Yes and no. Alternatives do exist, but they are much less popular and usually offer only a tiny fraction of the needed functionality.

So what does the future hold for core-js? After the announcement was made, a considerable amount of money was deposited into the core-js author's account. The question is how systematic the donations will be.

Discover more IT content selected for you

In Vived, you will find articles handpicked by devs. Download the app and read the good stuff!



## Criticism against Denis Pushkarev

In the end, I would like to mention the topic that core-js author tries to avoid as much as possible – politics. A few months ago, a statement by Denis Pushkarev appeared online, in which he blamed both sides for the war between Russia and Ukraine. He also repeatedly refers in his publications to the choice between the two evils. The developer himself states that he does not support Russia. He also can't leave the country even if he wanted to as he is still on probation.

My reporter's duty did not allow me to ignore the above facts as they are mentioned by countless people on Twitter and Reddit. Whether those facts change the perspective on the case I leave to your subjective judgment.